

EV251221695  
MAY 24 2003  
PATENT APPLICATION  
Docket No. MS1-230US  
5/20/03  
J. Mee



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of

Keith et al

Serial No.: 09/093,958

Confirmation No. 7840

Filed: June 8, 1998

For: Parcel Manager

Examiner: Boyce, A.

)  
)  
)  
)  
) Appeal No.  
)  
)  
)  
)  
)  
)  
)  
)

RECEIVED

MAY 16 2003

GROUP 3600

BRIEF OF APPELLANT

The Applicant has filed a timely Notice of Appeal from the action of the Examiner in finally rejecting all of the claims that were considered in this application. This Brief is being filed under the provisions of 37 C.F.R. § 1.192. The Filing Fee, as set forth in 37 C.F.R. § 1.17(c), is submitted herewith.

REAL PARTY IN INTEREST

The real party in interest comprises Microsoft Corporation, Inc. by way of assignment from Keith et al., who is the named inventive entity and is captioned in the present brief, to the

Microsoft Corporation, Inc. by way of an assignment document recorded at Reel/Frame 9421/0820 in the United States Patent and Trademark Office on August 24, 1998.

RELATED APPEALS AND INTERFERENCES

None.

STATUS OF CLAIMS

Claims 1-4, 20-23, and 36-42 are pending in the application.

STATUS OF AMENDMENTS

No amendment has been filed subsequent to the final Office action mailed on December 17, 2002 (hereinafter, the "FINAL").

SUMMARY OF INVENTION

This invention concerns a system and method for reliably transferring parcels from one computer to another and tracking the parcels as they are transferred. The system and method are described within the context of a distributed electronic billing system in which billers submit billing data to a service center and the service center generates billing statements from the billing data and electronically distributes the billing statements to consumers on behalf of the biller.

(Except from Specification, Page 19, lines 3-25, Page – Page 5, line 17)

## ISSUES

1. Whether Claims 1-4 and 36 satisfy the requirements of 35 U.S.C. § 102(e) as being unanticipated by Haff et al. (US Patent No. 6,219,669).
2. Whether Claims 20-23 and 37-42 satisfy the requirements of 35 U.S.C. § 103(a) as being nonobvious over Haff et al. (US Patent No. 6,219,669) in view of Kolling et al. (US Patent No. 5,963,925).

## GROUPING OF CLAIMS

There are two (2) separate grounds of rejection that are appealed herein:

- I. First Ground of Rejection: The grounds of the rejection based on 35 U.S.C. § 102(e) directed toward pending Claims 1-4 and 36 such that these claims stand or fall together as to this rejection.
- II. Second Ground of Rejection: The grounds of the rejection based on 35 U.S.C. 103(a) directed toward pending Claims 20-23 and 37-42 such that these claims stand or fall together as to this rejection.

## ARGUMENT

**First Ground of Rejection.** Claims 1-4 and 36 satisfy the requirements of 35 U.S.C. § 102(e) so as to be unanticipated by Haff et al. (US Patent No. 6,219,669).

1. **Data vs. File**

In order to show the distinction of the claim element “data” used in each independent

claim, the following explanation is given.

The recited claim element “data” is different from the term “file”. A “file” is a collection of data or information that has a name, called a filename. Almost all information stored in a computer must be in a file. There are many different types of files: data files, text files, program files, directory files, and so on. A program file contains a program. A data file contains data that can be limited to just one particular type of data.

All software is divided into two general categories: data and programs. Programs are collections of instructions for manipulating data. The content of a data file contains only data. The content of a program file is limited such that it must contain a program. As such, a data file is fundamentally different than a program file. It follows that the term ‘data’ is fundamentally different than the term ‘file’.

## 2. Haff et al. Teach File Packets Particularized By Files, Not By Data

Haff et al. teach at Col. 22, lines 10-16 a user interface by which a user can make a selection, by a “drag and drop feature”, of certain files from all files that are displayed on the user interface. Haff et al. do not teach that the user must select only certain files. Haff et al. do not teach that the user’s selection must be limited to only those files that contain a particular type of data. Haff et al. do not teach that the user must investigate what type of files are valid, based on one or more types of data they respectively contain, before the user makes a selection from among the displayed files. As such, Haff et al. teach that the user selects files, not data, and not particular data types. Haff et al. do not teach a file having different data types therein from which the user selects one of those data types for copying and subsequent transmission. Rather, the files that the user who is taught by Haff et al. selects are those files

that the user wishes to transmit. The user is not given a choice to transmit particular data types, but is given a choice to transmit particular files.

Haff et al. teach that the user-selected files are formed into a “file packet” that contains the user-selected files (Col. 22, lines 30-40). In contrast, Haff et al. do not teach that the file packet must be formed so that it will contain and carry only a particular type of data. Rather, the packets taught by Haff et al. contain only what happened to be in the files at the time that the user selected them. Haff et al. do not teach, suggest, or imply that a file packet is to contain only a particular type of data, but rather is to contain only the contents of those files that user happened to select – which selection is in no way limited by particular data type or by the type of data that is in each file. Neither is the user prompted to specify what type of data that the user wants to be in each file packet. Accordingly, Haff et al. disclose a file packet based on file selection, not on particular data type.

The FINAL, at paragraph 8 on Page 9, states that “Haff et al. disclose data specific to a credit authorization and data specific to a request for identification (i.e., types of data, see column 28, lines 6-10).” This specific passage from Haff et al. is seen in its intended context at Col. 28, lines 4-13, as follows:

Otherwise, at step 416 it is determined whether the *received data indicates a credit authorization*. If the data indicates a credit authorization, at step 418 transmission credits are added, described in greater detail with reference to FIG. 11. Otherwise, at step 420 it is determined whether the *received data indicates a request for identification*. If the received data indicates a request for identification, at step 422 a response is sent to the requesting PC with the ID of the receiving PC. Otherwise it is determined whether the received data is a partial file at step 424. (emphasis added)

The foregoing text teaches neither a file packet nor a relationship between a file packet and its payload. Neither here nor elsewhere do Haff et al. teach that the contents of a file

packet must exclude everything but data indicating credit authorization, everything but data indicating a request for identification, or everything but a particular kind of data. In contrast, Haff et al. teach credit in the context of a file transfer system as seen at Col. 6, lines 50-60:

In accordance with a preferred embodiment, the file transfer system of the invention also includes a credit sufficiency verifier that determines whether the local computer has *sufficient credits to transfer each selected file*. The credit sufficiency verifier allows the transmitter to operate only when sufficient credits are found. The sufficiency of credits is determined according to established transfer costs. Furthermore, the number of credits in the local machine is modified upon each successful file transfer by a corresponding transfer cost. A number of available credits for each local computer is displayed on the local computer. (emphasis added)

The foregoing supporting text from Haff et al. discloses a file transfer system. Haff et al. do not disclose that a file packet can include only credit data. With respect to a request for identification, Haff et al. teach a file transfer, not a file packet containing only a request for identification, as seen at Col. 37, lines 17-35:

In a preferred embodiment, a user function is provided to invoke a file request sequence that allows the user to request files from a selected index in order to request files from the associated destination PC address. Still further, when one or more files listed in the index are selected by the user, a control module initiates *transmission of the request for the files along with the identification and address of the requesting PC*. When a PC destination receives a request for one or more files in an index, a control module calls a compression subroutine which copies and compresses the files contained in the received request, creates a packet file and passes the packet to the pending events file. In another preferred embodiment, if automatic encryption is invoked, *packets containing the compressed files* are encrypted using a public key linked to the requesting destination or a one time key linked to the exchange transaction. A preferred compression algorithm (compress.d11) is publicly available from Infozip Group. Other compression algorithms are commercially available and may be used in preferred embodiments. (emphasis added)

3. **Applicant Discloses Parcel Components Particularized by Data, Not Files**

The specification, at page 18, lines 4-21, refers to Fig. 5 and sets forth the following:

The BIS gateway 80 has a parcel manager 134 to transfer billing data and other information from the BIS to the service center. The parcel manager transfers the data in "parcels". The parcel manager 134 is responsible for reliably transferring parcels from the BIS 34 to the service center and tracking the parcels as they go from computer to computer. It is this tracking function that enables the management console UI 100 to show the location and status of particular parcels. The parcel manager 134 is described below in more detail with reference to Fig. 7.

Atop the parcel manager 134 are a set of handlers that collectively form an enterprise interface into the parcel manager. The interface handlers handle *requests to create different types of parcels, depending upon the type of information being transferred* to the service center. The enterprise interface handlers include a consumer information handler 136, a payment handler 138, a batch handler 140, and a template handler 142. The handlers facilitate *creation of particularized parcels for shipment* to the service center. For instance, the batch handler 140 facilitates creation of statement batch parcel to be transferred to the service center. The handlers 136-142 are preferably implemented as COM (component object model) objects and are called via a set of enterprise integration APIs. (emphasis added)

The foregoing text presents the concept of particularization of the parcel contents. As designed, each parcel is to be limited as to its content. This limitation placed upon the content of the parcel is dictated by a request. The request is for a particular data type and is directed to a particular type of interface handler. The particular interface handler to which a particular request is directed is responsible for creating a parcel so as to contain the requested particular data type. As such, the disclosure provides for a particular type of interface handler to assemble a particular type of parcel to be made up of a particular type of data. Four (4) different types of data are disclosed: consumer information, payment, statement batch, and template.

3. **Claims 1-4 and 36 are not anticipated by Haff et al.**

The Applicant respectfully submits that the assembly of a file packet from user selected files taught by Haff et al. is fundamentally different than the assembly of a particular type of parcel to be made up of a particular type of data. The legal significance of this distinction is seen in MPEP § 2131, "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987).

Since Haff el al. do not teach the element of a "parcel component being particularized to contain and carry a particular type of data" recited in independent Claims 1-4, the Applicant respectfully asserts that the FINAL has not made out a case of anticipation with respect to these claims. In that Claim 36 depends from independent Claim 1, Claim 36 also includes the element missing from Haff et al. Accordingly, Claims 1-4 and 36 satisfy the requirements of 35 U.S.C. § 102(e) so as to be unanticipated by Haff et al. (US Patent No. 6,219,669). The Applicant respectfully requests the Board to overturn the First Ground of Rejection.

**Second Ground of Rejection.** Claims 20-23 and 37-42 satisfy the requirements of 35 U.S.C. § 103(a) so as to be nonobvious over Haff et al. (US Patent No. 6,219,669) in view of Kolling et al. (US Patent No. 5,963,925).

1. Claims 20-21 and 36 are dependent claims that depend directly from independent Claim 1. Claims 22-23 are dependent claims that depend directly from dependent Claim 36. Claims 37-38 are dependent claims that depend directly from independent Claim 2. Claims 39-40

are dependent claims that depend directly from independent Claim 3. Claims 41-42 are dependent claims that depend directly from independent Claim 4. The foregoing arguments presented under the First Ground of Rejection, above, in reference to the rejection of Claims 1-4 and 36 are hereby incorporated and directed to the rejection of Claims 20-23 and 37-42. Because Claims 20-23 and 37-42 incorporate all the limitations of the claims they respectively depend from, these claims are patentable for at least the same reasons for which Claims 1-4 and 36 are patentable.

The FINAL did not reject Claims 1-4 and 36 as being obvious over:

- (i) Haff et al. (US Patent No. 6,219,669); or
- (ii) Haff et al. (US Patent No. 6,219,669) in view of Kolling et al. (US Patent No. 5,963,925).

The Applicant agrees with (i) and (ii), above, and further states that “[i]f an independent claim is nonobvious under 35 U.S.C. § 103(a), then any claim depending therefrom is nonobvious.” (M.P.E.P. § 2143.03) (citing *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988)).

2. The Applicant respectfully asserts, in view of the foregoing, that the FINAL has not made out a *prima facie* case of obvious of Claims 20-23 and 37-42 under 35 U.S.C. § 103(a) over Haff et al. (US Patent No. 6,219,669) in view of Kolling et al. (US Patent No. 5,963,925), and that 20-23 and 37-42 satisfy the requirements of 35 U.S.C. § 103(a) so as to be nonobvious. The Applicant respectfully requests the Board to overturn the Second Ground of Rejection.

CONCLUSION

The Applicant respectfully considers this application to be in condition for allowance and respectfully request the Board to overturn the final rejection and that the Examiner pass this application to allowance.

Dated this 12<sup>th</sup> day of May, 2003.

Respectfully submitted,



BRADLEY K. DESANDRO  
Attorney for Applicant  
Registration No. 34,521

LEE & HAYES PLLC  
421 W. Riverside Avenue  
Suite 500  
Spokane, WA 99201  
Telephone: (509) 324-9256 (Ext. 228)  
Facsimile: (509) 323-8979

**APPENDIX: CLAIMS ON APPEAL**

1. (previously amended) A parcel manager for managing transfer of data from a local computer to a remote computer, the parcel manager being embodied on a computer readable medium, and comprising:

an interface object to present an interface into the parcel manager from one or more external applications;

a parcel object created via a first function presented by the interface object, the parcel object providing functionality to place the data in one or more parcel components for transferring to the remote computer, each said parcel component being particularized to contain and carry a particular type of data that was requested; and

a notification object created via a second function presented by the interface object in response to a request from an external application, the notification object providing functionality to track a status of the parcel object as the parcel components are transferred to the remote computer.

2. (previously amended) A parcel manager for managing transfer of data from a local computer to a remote computer, the parcel manager being embodied on a computer readable medium, and comprising:

an interface object to present an interface into the parcel manager from one or more external applications;

a parcel object created via a first function presented by the interface object, the parcel

object providing functionality to place the data in one or more parcel components for transferring to the remote computer, each said parcel component being particularized to contain and carry a particular type of data that was requested;

a notification object created via a second function presented by the interface object in response to a request from an external application, the notification object providing functionality to track a status of the parcel object as the parcel components are transferred to the remote computer; and

a bulletin object to hold update information regarding the parcel object.

3. (previously amended) A parcel manager for managing transfer of data from a local computer to a remote computer, the parcel manager being embodied on a computer readable medium, and comprising:

an interface object to present an interface into the parcel manager from one or more external applications;

a parcel object created via a first function presented by the interface object, the parcel object providing functionality to place the data in one or more parcel components for transferring to the remote computer, each said parcel component being particularized to contain and carry a particular type of data that was requested;

a notification object created via a second function presented by the interface object in response to a request from an external application, the notification object providing functionality to track a status of the parcel object as the parcel components are transferred to the remote computer; and

a monitor object created by the notification object to check for presence of the parcel components.

4. (previously amended) A parcel manager for managing transfer of data from a local computer to a remote computer, the parcel manager being embodied on a computer readable medium, and comprising:

an interface object to present an interface into the parcel manager from one or more external applications;

a parcel object created via a first function presented by the interface object, the parcel object providing functionality to place the data in one or more parcel components for transferring to the remote computer, each said parcel component being particularized to contain and carry a particular type of data that was requested;

a notification object created via a second function presented by the interface object in response to a request from an external application, the notification object providing functionality to track a status of the parcel object as the parcel components are transferred to the remote computer; and

a parcel database object to add and retrieve information regarding the parcel object in a database.

Claims 5-19 (canceled)

20. (original) The parcel manager as recited in claim 1, wherein the particular

type of data is batch statement data.

21. (original) The parcel manager as recited in claim 1, wherein the particular type of data is selected from the group consisting of consumer information data, payment data, batch statement data, and statement template data.

22. (previously amended) The parcel manager as recited in claim 36, wherein the particular type of electronic data is batch statement data.

23. (previously amended) The parcel manager as recited in claim 36, wherein the particular type of electronic data is selected from the group consisting of consumer information data, payment data, batch statement data, and statement template data.

Claims 24-35 (canceled)

36. (previously added) The parcel manager as defined in Claim 1, further comprising:

an electronic system executing on the local computer for submission of electronic data by a third party to the remote computer at a service center operated by a first party, wherein the service center generates an electronic statement from the electronic data and electronically distributes the electronic statements to a second party on behalf of the third party; and

a transfer and tracking object executing on the electronic computer system to manage the

transfer of the electronic data in said one or more parcel components to the service center operated by the first party and to track status of the electronic data as it is transferred.

37. (previously added) The parcel manager as recited in claim 2, wherein the particular type of data is batch statement data.

38. (previously added) The parcel manager as recited in claim 2, wherein the particular type of data is selected from the group consisting of consumer information data, payment data, batch statement data, and statement template data.

39. (previously added) The parcel manager as recited in claim 3, wherein the particular type of data is batch statement data.

40. (previously added) The parcel manager as recited in claim 3, wherein the particular type of data is selected from the group consisting of consumer information data, payment data, batch statement data, and statement template data.

41. (previously added) The parcel manager as recited in claim 4, wherein the particular type of data is batch statement data.

42. (previously added) The parcel manager as recited in claim 4, wherein the particular type of data is selected from the group consisting of consumer information data,

payment data, batch statement data, and statement template data.